

Con tali dati si costruiscono le matrici **A** e **B**:

$$\mathbf{A} = \begin{pmatrix} 2.3944 \cdot 10^{-8} & 1.5075 \cdot 10^{-9} & -7.2012 \cdot 10^{-10} \\ 1.5075 \cdot 10^{-9} & 2.5087 \cdot 10^{-8} & -9.8333 \cdot 10^{-10} \\ -7.2012 \cdot 10^{-10} & -9.8333 \cdot 10^{-10} & 2.7022 \cdot 10^{-10} \end{pmatrix} \quad (4.130)$$

$$\mathbf{B} = \begin{pmatrix} 2.3944 \cdot 10^{-8} & -1.5075 \cdot 10^{-9} & 7.2012 \cdot 10^{-10} \\ -1.5075 \cdot 10^{-9} & 2.5087 \cdot 10^{-8} & -9.8333 \cdot 10^{-10} \\ 7.2012 \cdot 10^{-10} & -9.8333 \cdot 10^{-10} & 2.7022 \cdot 10^{-10} \end{pmatrix} \quad (4.131)$$

Le dimensioni fisiche di **A** e **B** sono in tonnellate e metri.

4.8.1. Il caso del peso proprio

Le dividenti dell'arco sono 35; i valori delle quote e dei pesi propri sono quelli forniti dall'uscita del programma "AF" nel Capitolo 2. Poichè le stilate estreme scaricano, nel caso in esame, sulle imposte *A* e *B* dell'arco, occorrerà aggiungere le forze F_1 e F_{35} , pari a 10207 tonnellate ciascuna. Il file ASCII di preparazione dei dati ha il seguente aspetto:

```

"ARCO DI KRK - CARICO DA PESO PROPRIO",FILES,FILES
"ES4-10.ING", "ES4-10.USC"
400,34,4000000,1,0.00001,0,0
24,24,24,12.9,10,12.9,1.5,1.5,1.5,6,6,6
2.3944e-8,1.5075e-9,-7.2012e-10
1.5075e-9,2.5087e-8,-9.8333e-10
-7.2012e-10,-9.8333e-10,2.7022e-10
2.3944e-8,-1.5075e-9,7.2012e-10
-1.5075e-9,2.5087e-8,-9.8333e-10
7.2012e-10,-9.8333e-10,2.7022e-10
0,0,0,0,0,0
DATI,DATI
0,8.62083625695182,16.7973335025389,23.8464544250604,30.4889173145488
36.1927378977988,41.5181016707906,46.0549169977877,50.2342426106003
53.7423847761102,56.9084093786896,59.4923690477752,61.7451539054102
63.4800528277134,64.891089559284,65.82580314082,66.4408516752473
66.6,66.4408516752473,65.82580314082,64.891089559284,63.4800528277133
61.7451539054102,59.4923690477752,56.9084093786895,53.7423847761102
50.2342426106003,46.0549169977877,41.5181016707906,36.1927378977987
30.4889173145487,23.8464544250604,16.7973335025388,8.62083625695182,0
10207,4681.72368809505,11878.4628450262,4284.70266510642,9889.88994296598
3987.56392796541,8308.44432831668,3766.64668269305,7071.8423634285
3604.67988972197,6132.85603435181,3489.38292612778,5456.64186389326

```

3412.3343916476,5018.7206562288,3368.11170081178,4803.53684112721
 3353.69374670361,4803.53684112721,3368.11170081178,5018.72065622879
 3412.3343916476,5456.64186389326,3489.38292612777,6132.85603435181
 3604.67988972197,7071.8423634285,3766.64668269304,8308.44432831667
 3987.56392796541,9889.88994296597,4284.70266510642,11878.4628450262
 4681.72368809505,10207
 0,0

ed i risultati si ritrovano nel file "ES4-10.USC":

REAZIONE VERTICALE A SINISTRA = -101039.387620859
 REAZIONE ORIZZONTALE A SINISTRA = 118604.594001467
 REAZIONE VERTICALE A DESTRA = -101039.387620859
 REAZIONE ORIZZONTALE A DESTRA = -118604.594001467
 SOMMA DELLE FORZE VERTICALI = 202078.775241719
 SOMMA DELLE FORZE ORIZZONTALI = 0
 CARICO VERTICALE MEDIO = 505.196938104296
 CARICO ORIZZONTALE MEDIO = 0

| DIVIDENTE | MOMENTO FLETTENTE |
|-----------|-------------------|
| 1 | -257324.179560994 |
| 2 | -211178.638818607 |
| 3 | -167411.552247175 |
| 4 | -129679.076297873 |
| 5 | -94123.3560915556 |
| 6 | -63591.9926290677 |
| 7 | -35086.4296384079 |
| 8 | -10801.8022226088 |
| 9 | 11569.2572191198 |
| 10 | 30347.6114765898 |
| 11 | 47294.681683592 |
| 12 | 61126.0795471557 |
| 13 | 73184.7676295316 |
| 14 | 82471.3201769674 |
| 15 | 90024.3058786989 |
| 16 | 95027.6329775605 |
| 17 | 98319.8596662472 |
| 18 | 99171.7474976675 |
| 19 | 98319.8596662474 |
| 20 | 95027.6329775592 |
| 21 | 90024.3058786954 |
| 22 | 82471.3201769754 |

140 *Le strutture ad arco*

| | |
|----|-------------------|
| 23 | 73184.7676295295 |
| 24 | 61126.0795471504 |
| 25 | 47294.6816835992 |
| 26 | 30347.611476587 |
| 27 | 11569.2572191187 |
| 28 | -10801.802222615 |
| 29 | -35086.4296384137 |
| 30 | -63591.9926290616 |
| 31 | -94123.3560915487 |
| 32 | -129679.076297876 |
| 33 | -167411.55224717 |
| 34 | -211178.638818608 |
| 35 | -257324.179560995 |

| DIVIDENTE | N A SINISTRA | N A DESTRA |
|-----------|-------------------|-------------------|
| 1 | -155390.201785473 | -149357.163539941 |
| 2 | -149306.460665566 | -146586.270959108 |
| 3 | -146378.502601148 | -139925.301955276 |
| 4 | -139866.948497895 | -137711.993944766 |
| 5 | -137543.300826189 | -132948.087380643 |
| 6 | -132899.01810109 | -131206.589496034 |
| 7 | -131074.678379147 | -127862.964366391 |
| 8 | -127823.702440733 | -126515.288160595 |
| 9 | -126415.551838176 | -124219.304878204 |
| 10 | -124189.836904408 | -123206.16868276 |
| 11 | -123133.166503396 | -121677.295037004 |
| 12 | -121657.370030986 | -120954.777393746 |
| 13 | -120903.303298972 | -119991.531796027 |
| 14 | -119980.954100888 | -119528.741303575 |
| 15 | -119494.414466948 | -118996.54422273 |
| 16 | -118995.431374851 | -118774.070700427 |
| 17 | -118753.554660148 | -118595.587613817 |
| 18 | -118604.594001467 | -118604.594001467 |
| 19 | -118595.587613817 | -118753.554660148 |
| 20 | -118774.070700427 | -118995.431374851 |
| 21 | -118996.54422273 | -119494.414466948 |
| 22 | -119528.741303575 | -119980.954100888 |
| 23 | -119991.531796027 | -120903.303298972 |
| 24 | -120954.777393746 | -121657.370030986 |
| 25 | -121677.295037004 | -123133.166503396 |
| 26 | -123206.16868276 | -124189.836904408 |

4. Le sollecitazioni fisse 141

| | | |
|----|-------------------|-------------------|
| 27 | -124219.304878204 | -126415.551838176 |
| 28 | -126515.288160595 | -127823.702440733 |
| 29 | -127862.964366391 | -131074.678379147 |
| 30 | -131206.589496034 | -132899.01810109 |
| 31 | -132948.087380643 | -137543.300826189 |
| 32 | -137711.993944766 | -139866.948497895 |
| 33 | -139925.301955276 | -146378.502601148 |
| 34 | -146586.270959108 | -149306.460665566 |
| 35 | -149357.163539941 | -155390.201785473 |

| DIVIDENTE | TAGLIO A SINISTRA | TAGLIO A DESTRA |
|-----------|-------------------|-------------------|
| 1 | 11397.0504132016 | 3163.8676429757 |
| 2 | 5015.29288085581 | 1204.89858989967 |
| 3 | 7894.33914617195 | -2078.32754758617 |
| 4 | 4543.81893977835 | 840.462650907733 |
| 5 | 6865.86131138607 | -1891.64612360096 |
| 6 | 4077.16212508887 | 466.573323189744 |
| 7 | 5900.46656204619 | -1762.11396001063 |
| 8 | 3625.43742354156 | 93.3446580873116 |
| 9 | 5023.45548732023 | -1698.70589901935 |
| 10 | 3194.63860428545 | -273.230184622895 |
| 11 | 4249.46620944884 | -1708.0803838587 |
| 12 | 2786.75138172485 | -631.165643489067 |
| 13 | 3584.38221133927 | -1795.5446615104 |
| 14 | 2400.48670668728 | -981.750657302368 |
| 15 | 3027.99325813635 | -1965.97132375441 |
| 16 | 2032.21359879597 | -1328.61604296873 |
| 17 | 2576.49896544443 | -2224.43975488304 |
| 18 | 1676.84687335179 | -1676.84687335183 |
| 19 | 2224.439754883 | -2576.49896544448 |
| 20 | 1328.61604296865 | -2032.21359879604 |
| 21 | 1965.97132375486 | -3027.99325813589 |
| 22 | 981.750657302329 | -2400.48670668732 |
| 23 | 1795.54466150988 | -3584.3822113398 |
| 24 | 631.165643489505 | -2786.7513817244 |
| 25 | 1708.08038385867 | -4249.46620944887 |
| 26 | 273.230184622394 | -3194.63860428596 |
| 27 | 1698.70589901934 | -5023.45548732024 |
| 28 | -93.3446580873411 | -3625.43742354158 |
| 29 | 1762.11396001107 | -5900.46656204573 |
| 30 | -466.573323189284 | -4077.16212508841 |

142 *Le strutture ad arco*

| | | |
|----|-------------------|-------------------|
| 31 | 1891.64612360051 | -6865.86131138653 |
| 32 | -840.4626509077 | -4543.81893977832 |
| 33 | 2078.32754758615 | -7894.33914617197 |
| 34 | -1204.8985899001 | -5015.29288085626 |
| 35 | -3163.86764297573 | -11397.0504132016 |

| DIVIDENTE | SPOS. VERTICALI | SPOS. ORIZZONTALI |
|-----------|-----------------|-------------------|
|-----------|-----------------|-------------------|

| | | |
|----|----------------------|-----------------------|
| 1 | 2.42579495992211D-03 | -2.57008198738865D-03 |
| 2 | 7.45877294155403D-03 | -4.06276033220708D-03 |
| 3 | 1.57881989117706D-02 | -3.25444771207823D-03 |
| 4 | 2.62097270213975D-02 | -1.42202909606989D-03 |
| 5 | 3.89910387241859D-02 | 1.51804938153213D-03 |
| 6 | 5.28809306515876D-02 | 4.34182460829943D-03 |
| 7 | 6.80527391658789D-02 | 7.39580830069272D-03 |
| 8 | 8.34306114267813D-02 | 9.74727513152453D-03 |
| 9 | 9.91111583305062D-02 | 1.18084381614486D-02 |
| 10 | .114181909926326 | 1.29424923577102D-02 |
| 11 | .128688259032139 | 1.35357465520003D-02 |
| 12 | .141875599256271 | 1.32144368484639D-02 |
| 13 | .153770852423465 | 1.23082951980731D-02 |
| 14 | .163773917869628 | 1.06545596099982D-02 |
| 15 | .171930579938273 | 8.52514758966757D-03 |
| 16 | .177789851756042 | 5.91182187504344D-03 |
| 17 | .181454379218803 | 3.03472949124439D-03 |
| 18 | .182611226000799 | -9.34102535135186D-06 |
| 19 | .181454379218803 | -3.05341154194709D-03 |
| 20 | .177789851756042 | -5.93050392574615D-03 |
| 21 | .171930579938272 | -8.54382964037027D-03 |
| 22 | .163773917869628 | -1.06732416607009D-02 |
| 23 | .153770852423465 | -1.23269772487758D-02 |
| 24 | .141875599256271 | -1.32331188991666D-02 |
| 25 | .128688259032138 | -1.35544286027029D-02 |
| 26 | .114181909926326 | -1.29611744084128D-02 |
| 27 | 9.91111583305059D-02 | -1.18271202121513D-02 |
| 28 | .083430611426781 | -9.7659571822272D-03 |
| 29 | 6.80527391658788D-02 | -7.41449035139543D-03 |
| 30 | 5.28809306515877D-02 | -4.36050665900213D-03 |
| 31 | 3.89910387241861D-02 | -1.53673143223488D-03 |
| 32 | 2.62097270213977D-02 | 1.40334704536706D-03 |
| 33 | 1.57881989117708D-02 | 3.23576566137546D-03 |
| 34 | 7.45877294155417D-03 | 4.04407828150432D-03 |

35 2.42579495992218D-03 2.55139993668594D-03

| TRATTO | ROTAZIONE |
|--------|-----------------------|
| 1 | -2.56671681950422D-05 |
| 2 | -3.36557382091757D-04 |
| 3 | -5.99846536121494D-04 |
| 4 | -8.10339826263292D-04 |
| 5 | -9.76131699657125D-04 |
| 6 | -1.09578534796643D-03 |
| 7 | -1.17379060614189D-03 |
| 8 | -1.21102710919313D-03 |
| 9 | -1.21038119535158D-03 |
| 10 | -1.17418438800897D-03 |
| 11 | -1.10504010174519D-03 |
| 12 | -1.00626623233744D-03 |
| 13 | -8.80922177387795D-04 |
| 14 | -7.33059022207622D-04 |
| 15 | -5.66463868560407D-04 |
| 16 | -3.85720906490462D-04 |
| 17 | -1.95276544533312D-04 |
| 18 | 1.36880524276295D-18 |
| 19 | 1.95276544533315D-04 |
| 20 | 3.85720906490464D-04 |
| 21 | 5.66463868560404D-04 |
| 22 | 7.33059022207624D-04 |
| 23 | 8.80922177387802D-04 |
| 24 | 1.00626623233744D-03 |
| 25 | 1.10504010174519D-03 |
| 26 | 1.17418438800898D-03 |
| 27 | 1.21038119535159D-03 |
| 28 | 1.21102710919312D-03 |
| 29 | 1.17379060614188D-03 |
| 30 | 1.09578534796641D-03 |
| 31 | 9.76131699657122D-04 |
| 32 | 8.10339826263292D-04 |
| 33 | 5.99846536121495D-04 |
| 34 | 3.36557382091762D-04 |
| 35 | 2.56671681950458D-05 |

TEMPO DI ESECUZIONE = 7.97265625 SEC.

RUN DEL 09-29-1994 ORE 13:29:14

E' quindi:

$$H = 118604.594001467 t \quad M_A = -257324.179560994 tm \quad (4.132)$$

$$N_A = -155390.201785 t \quad N_C = -118604.59400 t \quad M_C = 99171.7474977 tm \quad (4.133)$$

La retta r (cfr. Figura 4.12) è definita da:

$$d = f \frac{M_A}{M_A - M_C} = 48.073 m \quad (4.134)$$

e di conseguenza, poichè:

$$M_A = -\Delta H \cdot d \quad (4.135)$$

la caduta di spinta sarà pari a:

$$\Delta H = -5352.79 t \quad (4.136)$$

ossia circa il 4.51% della spinta.

In Figura 4.21 sono riportati i diagrammi delle caratteristiche e degli spostamenti verticali.

4.8.2. Gli altri carichi

Il caso di una variazione termica uniforme di 25° conduce ai diagrammi di Figura 4.22, ed ai risultati seguenti. Sono riportati — per brevità — solo i valori delle caratteristiche e degli spostamenti agli estremi, alle reni ed in chiave, insieme ai diagrammi delle caratteristiche e degli spostamenti verticali:

REAZIONE VERTICALE A SINISTRA = 2.91044965905485D-13
 REAZIONE ORIZZONTALE A SINISTRA = 4410.46847838676
 REAZIONE VERTICALE A DESTRA = -2.91044965905485D-13
 REAZIONE ORIZZONTALE A DESTRA = -4410.46847838676
 SOMMA DELLE FORZE VERTICALI = 0
 SOMMA DELLE FORZE ORIZZONTALI = 0
 CARICO VERTICALE MEDIO = 0
 CARICO ORIZZONTALE MEDIO = 0

| DIVIDENTE | MOMENTO FLETTENTE |
|-----------|-------------------|
| 1 | 210638.601043881 |
| 8 | 7514.8413504196 |
| 9 | -10917.9425258048 |
| 18 | -83098.5996166774 |

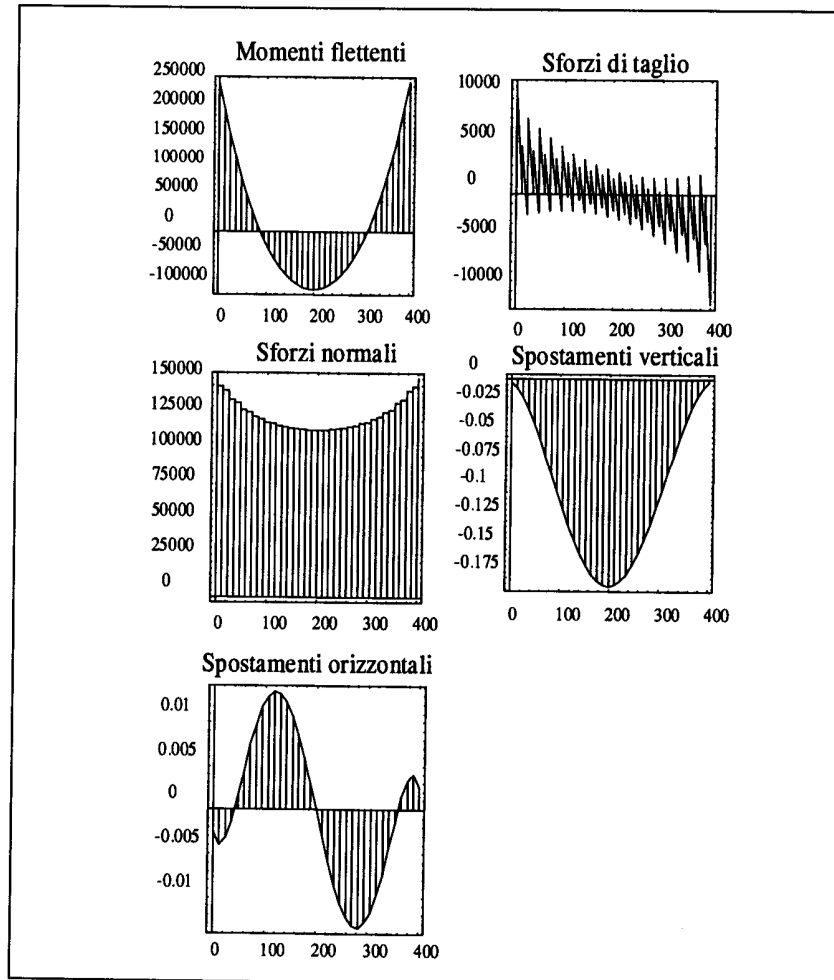


Figura 4.21- L'arco di esempio soggetto al peso proprio

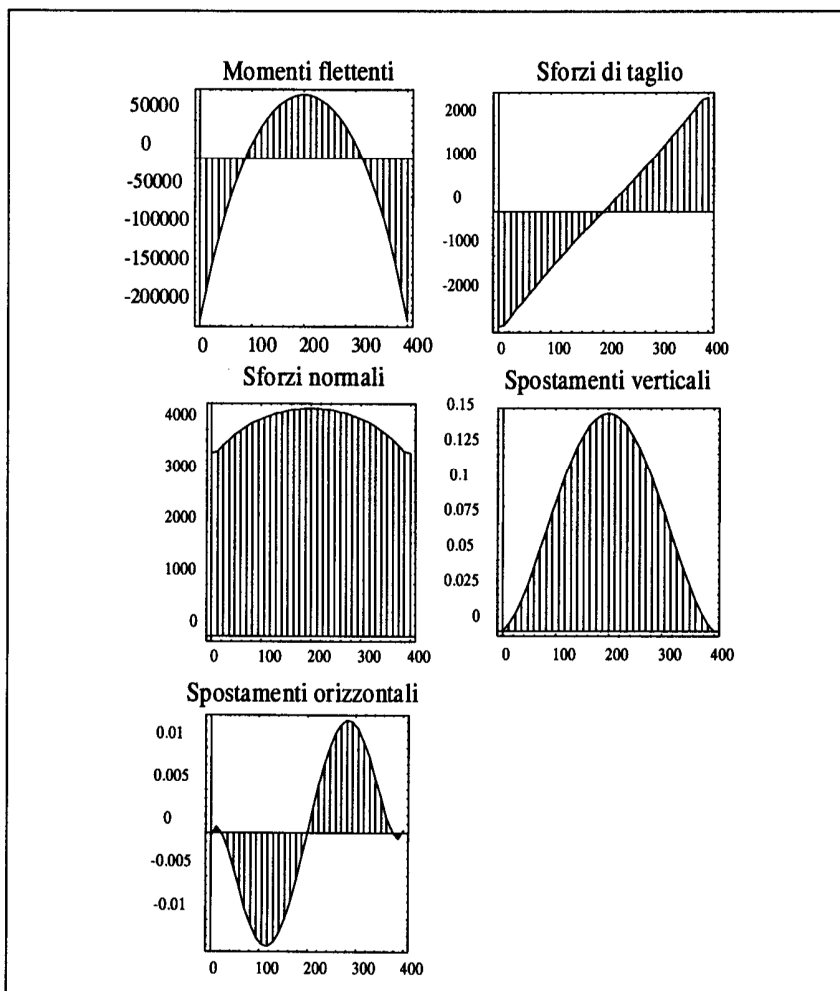


Figura 4.22- L'arco di esempio soggetto ad una variazione termica

| | |
|----|-------------------|
| 27 | -10917.9425258049 |
| 28 | 7514.84135041952 |
| 35 | 210638.601043881 |

| DIVIDENTE | N A SINISTRA | N A DESTRA |
|-----------|-------------------|-------------------|
| 1 | -3557.57745516589 | -3557.57745516589 |
| 9 | -4192.38430064578 | -4192.38430064578 |
| 18 | -4410.46847838676 | -4410.46847838676 |
| 27 | -4192.38430064578 | -4192.38430064578 |
| 35 | -3557.57745516589 | -3557.57745516589 |

| DIVIDENTE | TAGLIO A SINISTRA | TAGLIO A DESTRA |
|-----------|-----------------------|-----------------------|
| 1 | -2606.88988055472 | -2606.88988055472 |
| 9 | -1369.72481708628 | -1369.72481708628 |
| 18 | -2.91044965905485D-13 | -2.91044965905485D-13 |
| 27 | 1369.72481708628 | 1369.72481708628 |
| 35 | 2606.88988055472 | 2606.88988055472 |

| DIVIDENTE | SPOS. VERTICALI | SPOS. ORIZZONTALI |
|-----------|-----------------------|-----------------------|
| 1 | -1.58333850614888D-04 | -3.17772678281768D-04 |
| 9 | -8.34482205759377D-02 | -1.21661831445099D-02 |
| 18 | -.154039211294731 | 8.46222321145286D-06 |
| 27 | -8.34482205759377D-02 | 1.21831075909328D-02 |
| 35 | -1.58333850614902D-04 | 3.34697124704685D-04 |

| TRATTO | ROTAZIONE |
|--------|-----------------------|
| 1 | 6.12557087429296D-05 |
| 9 | 1.02075486811533D-03 |
| 18 | -5.14996031930615D-19 |
| 27 | -1.02075486811533D-03 |
| 35 | -6.12557087429296D-05 |

TEMPO DI ESECUZIONE = 7.63671875 SEC.

RUN DEL 09-29-1994 ORE 15:18:47

Un sismo sussultorio può essere simulato (se l'accelerazione sismica si assume pari ad un decimo della accelerazione di gravità) da un carico orizzontale uniformemente distribuito lungo z , pari a 50 tm^{-1} . Si hanno quindi le forze orizzontali seguenti:

$$P_1 = 0 \quad P_2 = P_3 = \dots = P_{34} = 588t \quad P_{35} = 0 \quad (4.137)$$

che conducono ai diagrammi di Figura 4.23, ed ai risultati seguenti:

REAZIONE VERTICALE A SINISTRA = 1277.70412235189
 REAZIONE ORIZZONTALE A SINISTRA = -9705.88235294117
 REAZIONE VERTICALE A DESTRA = -1277.70412235189
 REAZIONE ORIZZONTALE A DESTRA = -9705.88235294118
 SOMMA DELLE FORZE VERTICALI = 0
 SOMMA DELLE FORZE ORIZZONTALI = 19411.7647058824
 CARICO VERTICALE MEDIO = 0
 CARICO ORIZZONTALE MEDIO = 48.5294117647059

| DIVIDENTE | MOMENTO FLETTENTE |
|-----------|-----------------------|
| 1 | -203035.443169109 |
| 9 | 77148.05288376 |
| 10 | 79656.9505071697 |
| 11 | 78592.8929026448 |
| 18 | -1.13161181813926D-10 |
| 25 | -78592.8929026446 |
| 26 | -79656.9505071698 |
| 27 | -77148.0528837602 |
| 35 | 203035.443169109 |

| DIVIDENTE | N A SINISTRA | N A DESTRA |
|-----------|-------------------|-------------------|
| 1 | 8584.18156115391 | 8584.18156115391 |
| 9 | 5708.72075926542 | 5149.57190144462 |
| 18 | 294.117647058823 | -294.117647058824 |
| 19 | -335.976593865658 | -923.893725262099 |
| 27 | -5149.57190144463 | -5708.72075926543 |
| 35 | -8584.18156115391 | -8584.18156115391 |

| DIVIDENTE | TAGLIO A SINISTRA | TAGLIO A DESTRA |
|-----------|-------------------|-------------------|
| 1 | 4706.22003295162 | 4706.22003295162 |
| 9 | 520.969115619299 | 338.285465438514 |
| 10 | 135.2216773171 | -25.2997513194614 |
| 11 | -193.876960910525 | -333.517444142781 |
| 18 | -1277.70412235189 | -1277.70412235189 |
| 25 | -333.517444142778 | -193.876960910522 |
| 26 | -25.2997513194774 | 135.221677317082 |
| 27 | 338.285465438519 | 520.969115619305 |

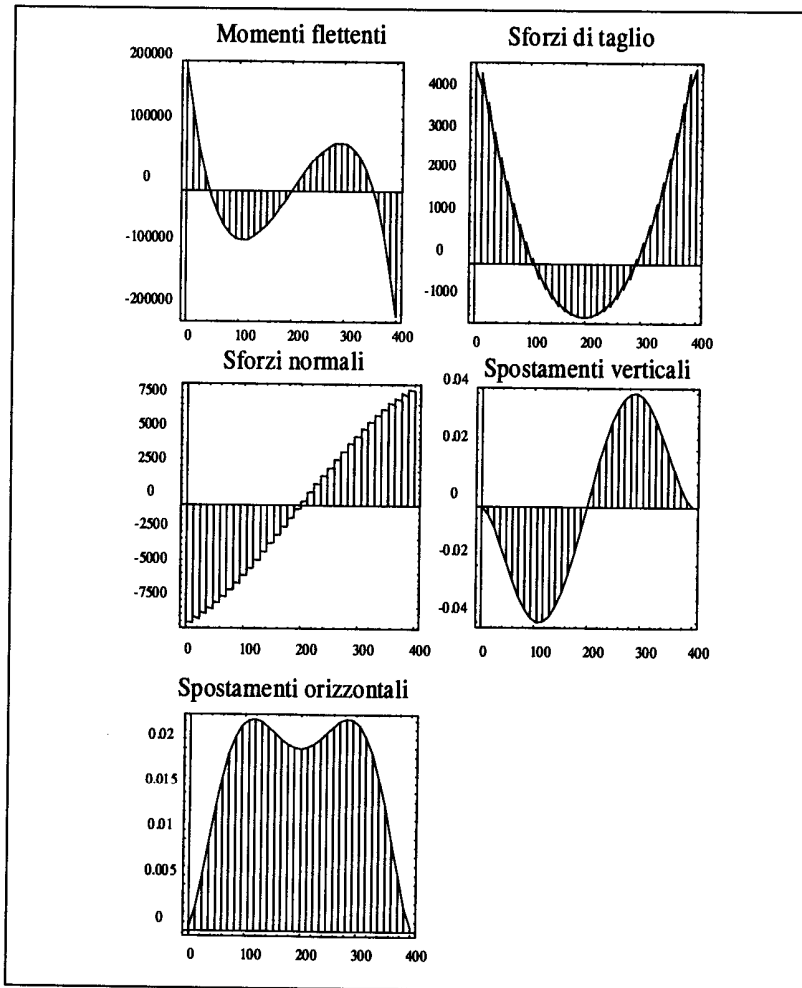


Figura 4.23- L'arco di esempio soggetto al carico orizzontale sismico

| DIVIDENTE | SPOS. VERTICALI | SPOS. ORIZZONTALI |
|-----------|-----------------------|----------------------|
| 1 | 1.30248153476404D-04 | 4.4121617395527D-04 |
| 9 | 3.83664255957967D-02 | 2.24957038534346D-02 |
| 18 | 1.69759477163184D-07 | 1.99253218397648D-02 |
| 27 | -3.83660860768422D-02 | 2.24957038534346D-02 |
| 35 | -1.29908634521849D-04 | 4.41216173955223D-04 |

| TRATTO | ROTAZIONE |
|--------|-----------------------|
| 1 | -6.34882224546864D-05 |
| 9 | -1.95128570712657D-04 |
| 18 | 6.89010792310446D-04 |
| 27 | -1.95128570712659D-04 |
| 35 | -6.34882224546867D-05 |

TEMPO DI ESECUZIONE = 7.69140625 SEC.

RUN DEL 09-29-1994 ORE 15:24:18

Infine, in Figura 4.24 sono riportati i diagrammi relativi ad un carico accidentale uniforme, di intensità pari a 60 tm^{-1} , esteso da $z = 0$ a $z = l/2$. Tale carico si trasforma quindi nelle forze verticali:

$$\begin{aligned} F_1 &= 353 t & F_2 = F_3 = \dots = F_{17} &= 706 t \\ F_{18} &= 353 t & F_{19} = F_{20} = \dots = F_{35} &= 0 \end{aligned} \quad (4.138)$$

che conducono ai seguenti risultati:

REAZIONE VERTICALE A SINISTRA = -9780.15482795453
 REAZIONE ORIZZONTALE A SINISTRA = 8808.44175585523
 REAZIONE VERTICALE A DESTRA = -2221.84517204547
 REAZIONE ORIZZONTALE A DESTRA = -8808.44175585523
 SOMMA DELLE FORZE VERTICALI = 12002
 SOMMA DELLE FORZE ORIZZONTALI = 0
 CARICO VERTICALE MEDIO = 30.005
 CARICO ORIZZONTALE MEDIO = 0

| DIVIDENTE | MOMENTO FLETTENTE |
|-----------|-------------------|
| 1 | -152838.172990948 |
| 9 | 59373.3518080334 |
| 18 | 16350.5716599986 |
| 19 | -8386.93455611623 |

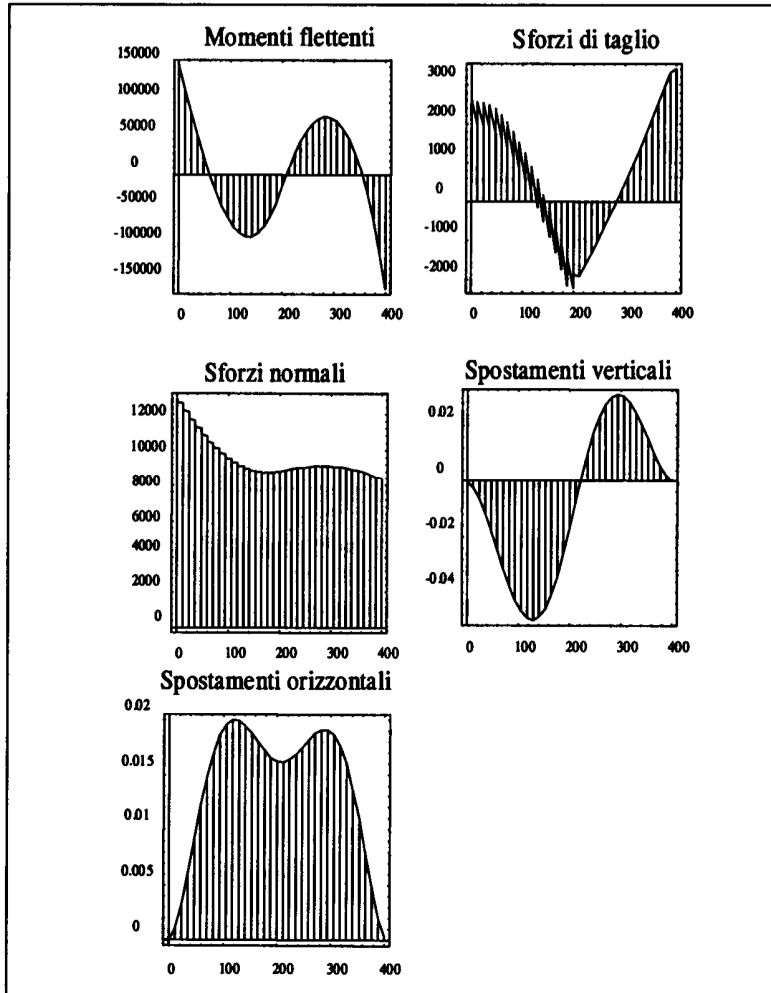


Figura 4.24- L'arco di esempio soggetto ad un carico su metà luce

27 -74746.8022721845
 35 158623.758190863

| DIVIDENTE | N A SINISTRA | N A DESTRA |
|-----------|-------------------|-------------------|
| 1 | -12885.8194397103 | -12677.1721877584 |
| 9 | -9765.81082435906 | -9546.55390741208 |
| 18 | -8808.44175585523 | -8808.44175585523 |
| 27 | -9062.91238632656 | -9062.91238632656 |
| 35 | -8418.33915895436 | -8418.33915895436 |

| DIVIDENTE | TAGLIO A SINISTRA | TAGLIO A DESTRA |
|-----------|-------------------|-------------------|
| 1 | 2682.48615881633 | 2397.74886892804 |
| 9 | 1527.80854162072 | 856.718082464198 |
| 18 | -1868.84517204547 | -2221.84517204547 |
| 27 | 623.587373673032 | 623.587373673032 |
| 35 | 3414.20678044715 | 3414.20678044715 |

| DIVIDENTE | SPOS. VERTICALI | SPOS. ORIZZONTALI |
|-----------|-----------------------|-----------------------|
| 1 | 3.30959126387833D-04 | -5.59434342788094D-05 |
| 9 | 4.25695274791991D-02 | 1.85608613875276D-02 |
| 18 | 1.73467152052473D-02 | 1.62184686516558D-02 |
| 27 | -3.04391582115273D-02 | 1.86230082206373D-02 |
| 35 | -7.36680758695299D-05 | 3.7218403320977D-04 |

| TRATTO | ROTAZIONE |
|--------|-----------------------|
| 1 | -3.96812111685356D-05 |
| 9 | -3.8781495805415D-04 |
| 18 | 7.6646886595978D-04 |
| 27 | -1.27512304532854D-04 |
| 35 | -4.99249218248262D-05 |

TEMPO DI ESECUZIONE = 8.0234375 SEC.

RUN DEL 09-29-1994 ORE 15:35:37

In Tabella 4.1 sono riportati i valori delle sollecitazioni interne nella sezione *A* di imposta nei quattro casi esaminati, e nel caso in cui il carico accidentale insista sulla semiluce di destra.

| z | g | Δt | s | a_s | a_d |
|-------------------------|---------|--------------|--------------|---------|--------|
| M (tm) | -257324 | ± 210638 | ∓ 203035 | -152838 | 158623 |
| N (t) | -155390 | ∓ 3557 | ± 8584 | -12885 | -8418 |
| T (t) | 3163 | ∓ 2606 | ± 4706 | 2682 | -3414 |
| v (cm) | 0.243 | ∓ 0.016 | ± 0.013 | 0.033 | 0 |
| w (cm) | -0.257 | ∓ 0.032 | ± 0.044 | -0.006 | -0.037 |
| $\phi \cdot 10^5$ (rad) | -2.57 | ± 6.13 | ∓ 6.34 | -3.96 | 5 |

Tabella 4.1- Caratteristiche per varie condizioni di carico in $z = 0$

Sotto $\Delta t = -25^\circ$, il sismo verso destra ed il carico accidentale sulla semiluce di sinistra si ha il massimo momento negativo:

$$M = -823835 \text{ tm} \quad (4.139)$$

accompagnato da:

$$N = -156134 \text{ t}; \quad (4.140)$$

l'eccentricità è quindi:

$$e = \frac{M}{N} = 5.276 \text{ m}. \quad (4.141)$$

La sezione è parzializzata, essendo all'imposta:

$$b = 12.92 \text{ m} \quad A = 131.52 \text{ m}^2 \quad I = 2849 \text{ m}^4 \quad (4.142)$$

$$\rho_x = \sqrt{\frac{I}{A}} = 4.65 \text{ m} \quad r_x = \frac{(4.65)^2}{6.46} = 3.35 \text{ m}$$

Un calcolo eseguito nell'ipotesi di sezione tutta reagente porterebbe ai seguenti valori delle tensioni:

$$\sigma = -\frac{156134}{131.52} \pm \frac{823835}{2849} \frac{12.92}{2} \quad (4.143)$$

e quindi:

$$\sigma_e = 680.86 \text{ tm}^{-2} \quad \sigma_i = -3055 \text{ tm}^{-2} \quad (4.144)$$

Inoltre, può porsi, nello stesso ordine di approssimazione in cui si scrive la (4.97):

$$\tau = \frac{T}{c(b-2s)} \quad (4.145)$$

e quindi $\tau = 222 \text{ tm}^{-2}$.

Con una armatura simmetrica di 1 m^2 , ed assumendo pari a 15 il coefficiente di amplificazione delle aree, si ha invece:

$$\begin{aligned} \sigma_{ci} &= -2092 \text{ tm}^{-2} & \sigma_{fi} &= -27286 \text{ tm}^{-2} \\ \sigma_{fe} &= -180 \text{ tm}^{-2} & \tau_{max} &= 243 \text{ tm}^{-2} \end{aligned} \quad (4.146)$$

I valori delle tensioni sono piuttosto elevati; si tenga conto però che le tecniche di esecuzione portano sempre ad una riduzione delle sollecitazioni interne da peso proprio. Si tenga anche conto, poi, che le variazioni termiche — cui si deve gran parte del quadro tensionale — sono stagionali, e la loro lentezza chiama in gioco la cedibilità elastica lenta delle fondazioni. Ciò autorizza, come sovente si pratica, a considerare, nel calcolo degli effetti termici, un modulo elastico ridotto, a volte addirittura dimezzato. Nel caso in esame esiste già — a prescindere dalla deformabilità del terreno — una cedibilità elastica del blocco di fondazione, definito dalle matrici (4.130) e (4.131), e dovuto alla deformabilità del telaio della Figura 4.20. Le cedibilità sono però molto basse, come può verificarsi calcolando le sollecitazioni da variazione termica in loro assenza. Si ha infatti, per $\Delta t = 25^\circ$:

| C.s.i. | Cedib. reali | Cedib. nulle |
|--------|------------------|--------------|
| M_A | 210638 <i>tm</i> | 227284 |
| M_C | -83098 <i>tm</i> | -86653 |
| N_A | -3557 <i>t</i> | -3806 |
| N_C | -4410 <i>t</i> | -4719 |
| T_A | -2606 <i>t</i> | -2789 |

Tabella 4.2- Caratteristiche per vincoli perfetti e cedevoli

Gli spostamenti v_A, w_A, ϕ_A sono forniti dalle ultime tre colonne della Tabella 4.1, e rappresentano appunto l'effetto della cedibilità dei due telai di fondazione.

Appendice 4.1. Il programma ADL

```
DEFDBL A-Z
DECLARE SUB Plv1 (MA(), MB(), NA(), NB(), TA(), TB(), C)
DECLARE SUB Caratteristiche ()
DECLARE SUB Cedibilita ()
DECLARE SUB TerminiNoti ()
DECLARE SUB Geometria ()
DECLARE SUB Spostamenti ()
DECLARE SUB Iperstatiche ()
DECLARE SUB slsim (A(), B(), X(), N)
DECLARE SUB Imposte (A(), B(), RA(), RB(), PA(), PB(), C)
DECLARE SUB Imposte1 (SA(), SB(), RA(), RB(), C)
DECLARE SUB product (A(), B(), C(), N)
DECLARE SUB dot (A(), B(), C, N)
DECLARE SUB Plv (MA(), MB(), NA(), NB(), TA(), TB(), C)
DECLARE SUB UscitaDati ()
DECLARE SUB UscitaRisultati ()
DECLARE SUB IngressoDati ()
DECLARE SUB IngressoDatiArray ()
DECLARE SUB OpenFiles ()
REM *****
REM *
REM *          P R O G R A M M A      A D L          *
REM *
REM *
REM *          di
REM *
REM *          V I N C E N Z O      F R A N C I O S I
REM *
REM *          Release 1.0.0. Gennaio 1988 (in BASIC HP)
REM *          Release 3.0.0. Agosto 1994 (in Microsoft QBasic)
REM *
REM *****
REM
```

```

REM
REM *****
REM *
REM *           Questo programma studia l'arco in fase elastica per il caso
REM *                   di sollecitazioni fisse
REM *
REM *****
OPTION BASE 1
REM
REM *****
REM *
REM *                   C O M M O N
REM *
REM *****
REM
COMMON SHARED LUCE, T, E, NU, G, FREC, RIAL
COMMON SHARED ALPHA, DELTATN, DELTATU
COMMON SHARED LARS, LARC, LARD, ALTS, ALTC, ALTD
COMMON SHARED SPES, SPEC, SPED, SPTS, SPTC, SPTD
COMMON SHARED FVS, FVC, FVD, FOS, FOC, FOD
COMMON SHARED ITER1, ITER2, TIME2
COMMON SHARED M() AS DOUBLE, M0() AS DOUBLE, M1() AS DOUBLE
COMMON SHARED M2() AS DOUBLE, M3() AS DOUBLE
COMMON SHARED TG() AS DOUBLE, T0() AS DOUBLE, T1() AS DOUBLE
COMMON SHARED T2() AS DOUBLE, T3() AS DOUBLE
COMMON SHARED N() AS DOUBLE, N0() AS DOUBLE, N1() AS DOUBLE
COMMON SHARED N2() AS DOUBLE, N3() AS DOUBLE
COMMON SHARED X() AS DOUBLE, INCL() AS DOUBLE
COMMON SHARED FVER() AS DOUBLE, FORI() AS DOUBLE
COMMON SHARED R0() AS DOUBLE, P0() AS DOUBLE
COMMON SHARED R() AS DOUBLE, P() AS DOUBLE, INERZ() AS DOUBLE
COMMON SHARED R1() AS DOUBLE, R2() AS DOUBLE, R3() AS DOUBLE
COMMON SHARED P1() AS DOUBLE, P2() AS DOUBLE, P3() AS DOUBLE
COMMON SHARED C() AS DOUBLE, AREA() AS DOUBLE
COMMON SHARED CHI() AS DOUBLE, Q() AS DOUBLE
COMMON SHARED SA() AS DOUBLE, SB() AS DOUBLE
COMMON SHARED A() AS DOUBLE, B() AS DOUBLE, BB() AS DOUBLE
COMMON SHARED V() AS DOUBLE, W() AS DOUBLE, PHI() AS DOUBLE
COMMON SHARED NS() AS DOUBLE, TS() AS DOUBLE
COMMON SHARED TITLE$, STAMPA$, STAMPARIS$, FILEINGR$, FILEUSC$
COMMON SHARED ASSE$, CARICHI$, FILEINPUT$
REM

```

```

REM *****
REM *
REM *          DIMENSIONA GLI ARRAY STATICI          *
REM *
REM *****
REM
DIM TITLE$(100), TITLE1$(100), STAMPA$(10), STAMPARIS$(10)
DIM FILEINGR$(10), FILEUSC$(10)
DIM R0(3) AS DOUBLE, R1(3) AS DOUBLE, R2(3) AS DOUBLE
DIM P0(3) AS DOUBLE, P1(3) AS DOUBLE, P2(3) AS DOUBLE
DIM P(3) AS DOUBLE, R(3) AS DOUBLE
DIM C(3, 3) AS DOUBLE, BB(3) AS DOUBLE, X(3) AS DOUBLE
DIM SA(3) AS DOUBLE, SB(3) AS DOUBLE, R3(3) AS DOUBLE
DIM A(3, 3) AS DOUBLE, B(3, 3) AS DOUBLE, P3(3) AS DOUBLE
REM
REM *****
REM *
REM *          DEFINISCE ALCUNE LEGGI DI VARIAZIONE          *
REM *
REM *****
REM
DEF FNA (Z) = LARS + (LARD - LARS) / LUCE * Z - 4 * (LARS + (LARD -
  LARS) / 2 - LARC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNB (Z) = ALTS + (ALTD - ALTS) / LUCE * Z - 4 * (ALTS + (ALTD -
  ALTS) / 2 - ALTC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNC (Z) = SPTS + (SPTD - SPTS) / LUCE * Z - 4 * (SPTS + (SPTD - SPTS)
  / 2 - SPTC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNS (Z) = SPES + (SPED - SPES) / LUCE * Z - 4 * (SPES + (SPED - SPES)
  / 2 - SPEC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNQ (Z) = RIAL * Z / LUCE + 4 * FREC / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNF (Z) = FVS + (FVD - FVS) / LUCE * Z - 4 * (FVS + (FVD - FVS) / 2 -
  FVC) / LUCE ^ 2 * Z * (LUCE - Z)
DEF FNP (Z) = FOS + (FOD - FOS) / LUCE * Z - 4 * (FOS + (FOD - FOS) / 2 -
  FOC) / LUCE ^ 2 * Z * (LUCE - Z)
CLS
REM
REM *****
REM *
REM *          LEGGE IL NOME DEL FILE DI INPUT DATI          *
REM *
REM *****
REM

```

```

PRINT "NOME DEL FILE DI INPUT"
INPUT FILEINPUT$
TIME1 = TIMER
OPEN FILEINPUT$ FOR INPUT AS #7
REM -----
REM CHIAMA LA SUBROUTINE PER L'INGRESSO DEI DATI
REM -----
CALL IngressoDati
G = E / (2 * (1 + NU))
REM
REM *****
REM *
REM *          REDIMENSIONA GLI ARRAY DINAMICI          *
REM *
REM *****
REM
REDIM M(T + 1) AS DOUBLE, M0(T + 1) AS DOUBLE, M1(T + 1) AS DOUBLE
REDIM M2(T + 1) AS DOUBLE, M3(T + 1) AS DOUBLE
REDIM TG(T + 1) AS DOUBLE, T0(T + 1) AS DOUBLE, T1(T + 1) AS DOUBLE
REDIM T2(T + 1) AS DOUBLE, T3(T + 1) AS DOUBLE
REDIM N(T + 1) AS DOUBLE, N0(T + 1) AS DOUBLE, N1(T + 1) AS DOUBLE
REDIM N2(T + 1) AS DOUBLE, N3(T + 1) AS DOUBLE
REDIM FVER(T + 1) AS DOUBLE, FORI(T + 1) AS DOUBLE
REDIM AREA(T + 1) AS DOUBLE, INERZ(T + 1) AS DOUBLE
REDIM CHI(T + 1) AS DOUBLE, Q(T + 1) AS DOUBLE
REDIM INCL(T + 1) AS DOUBLE, PHI(T + 1) AS DOUBLE
REDIM V(T + 1) AS DOUBLE, W(T + 1) AS DOUBLE
REDIM NS(T + 1) AS DOUBLE, TS(T + 1) AS DOUBLE
REM -----
REM CHIAMA LA SUBROUTINE PER L'INGRESSO DEI DATI IN ARRAY
REM -----
CALL IngressoDatiArray
REM -----
REM CHIAMA LA SUBROUTINE PER RIEMPIRE ALCUNI ARRAY
REM -----
CALL Geometria
REM -----
REM CHIAMA LA SUBROUTINE PER STAMPARE I DATI DI INGRESSO
REM -----
TIME2 = TIMER - TIME1
CALL UscitaDati
TIME1 = TIMER

```

```

REM -----
REM CHIAMA LA SUBROUTINE PER RIEMPIRE LA MATRICE
REM DELLE FLESSIBILITA'
REM -----
CALL Cedibilita
REM -----
REM CHIAMA LA SUBROUTINE PER CALCOLARE I TERMINI
REM NOTI DELLE EQUAZIONI DI CONGRUENZA
REM -----
CALL TerminiNoti
REM -----
REM CHIAMA LA SUBROUTINE PER CALCOLARE LE IPERSTATICHE
REM -----
CALL Iperstatiche
REM -----
REM CHIAMA LA SUBROUTINE PER CALCOLARE LE CARATTERISTICHE
REM -----
CALL Caratteristiche
REM -----
REM CHIAMA LA SUBROUTINE PER CALCOLARE GLI SPOSTAMENTI
REM -----
CALL Spostamenti
REM -----
REM CHIAMA LA SUBROUTINE PER LA STAMPA DEI RISULTATI
REM -----
TIME2 = TIME2 + TIMER - TIME1
CALL UscitaRisultati
REM -----
REM CHIUDE TUTTI I BUFFER TEMPORANEI
REM -----
CLOSE END
SUB Caratteristiche
REM
REM *****
REM *                                     *
REM *               C a r a t t e r i s t i c h e               *
REM *                                     *
REM *****
REM
REM *****
REM *                                     *
REM *   QUESTA SUBROUTINE CALCOLA LE CARATTERISTICHE,         *

```

```

REM *          NOTE LE INCOGNITE IPERSTATICHE          *
REM *          *          *          *          *          *
REM *****
REM Cfr. (4.47)
FOR I = 1 TO T + 1
  M(I) = M0(I) + X(1) * M1(I) + X(2) * M2(I) + X(3) * M3(I)
  N(I) = N0(I) + X(1) * N1(I) + X(2) * N2(I) + X(3) * N3(I)
  TG(I) = T0(I) + X(1) * T1(I) + X(2) * T2(I) + X(3) * T3(I)
NEXT I
REM Cfr. (4.49)
FOR I = 1 TO T + 1
  NS(I) = N(I) - FVER(I) * SIN(INCL(I)) + FORI(I) * COS(INCL(I))
  TS(I) = TG(I) + FVER(I) * COS(INCL(I)) + FORI(I) * SIN(INCL(I))
NEXT I
END SUB
SUB Cedibilita
REM
REM *****
REM *          *          *          *          *          *
REM *          Cedibilita          *
REM *          *          *          *          *          *
REM *****
REM
REM *****
REM *          *          *          *          *          *
REM *          QUESTA SUBROUTINE CALCOLA I TERMINI          *
REM *          DELLA MATRICE DI CEDIBILITA'          *
REM *          *          *          *          *          *
REM *****
REM
REM -----
REM Momenti, tagli e sforzi normali
REM -----
REM
FOR I = 1 TO T + 1
  Z = LUCE / T * (I - 1)
  REM Cfr. (4.7)
  M1(I) = (RIAL / 2 / FREC / LUCE - 1 / LUCE) * Z - Q(I) / 2 / FREC + 1
  N1(I) = (1 / LUCE - RIAL / 2 / FREC / LUCE) * SIN(INCL(I))
  N1(I) = N1(I) - 1 / 2 / FREC * COS(INCL(I))
  T1(I) = (RIAL / 2 / FREC / LUCE - 1 / LUCE) * COS(INCL(I))
  T1(I) = T1(I) - 1 / 2 / FREC * SIN(INCL(I))

```

```

REM Cfr. (4.10)
M2(I) = -RIAL / FREC / LUCE * Z + Q(I) / FREC
N2(I) = RIAL / FREC / LUCE * SIN(INCL(I)) + COS(INCL(I)) / FREC
T2(I) = -RIAL / FREC / LUCE * COS(INCL(I)) + SIN(INCL(I)) / FREC
REM Cfr. (4.13)
M3(I) = (RIAL / 2 / FREC / LUCE + 1 / LUCE) * Z - Q(I) / 2 / FREC
N3(I) = -(1 / LUCE + RIAL / 2 / FREC / LUCE) * SIN(INCL(I))
N3(I) = N3(I) - 1 / 2 / FREC * COS(INCL(I))
T3(I) = (RIAL / 2 / FREC / LUCE + 1 / LUCE) * COS(INCL(I))
T3(I) = T3(I) - 1 / 2 / FREC * SIN(INCL(I))
NEXT I
REM -----
REM Matrici delle reazioni
REM -----
REM
REM Cfr. (4.6)
R1(1) = 1 / LUCE - RIAL / 2 / FREC / LUCE
R1(2) = 1 / FREC / 2
R1(3) = -1
P1(1) = -1 / LUCE + RIAL / 2 / FREC / LUCE
P1(2) = -1 / FREC / 2
P1(3) = 0
REM Cfr. (4.9)
R2(1) = RIAL / FREC / LUCE
R2(2) = -1 / FREC
R2(3) = 0
P2(1) = -RIAL / FREC / LUCE
P2(2) = 1 / FREC
P2(3) = 0
REM Cfr. (4.12)
R3(1) = -RIAL / 2 / FREC / LUCE - 1 / LUCE
R3(2) = 1 / 2 / FREC
R3(3) = 0
P3(1) = RIAL / 2 / FREC / LUCE + 1 / LUCE
P3(2) = -1 / 2 / FREC
P3(3) = 1
REM -----
REM Costruzione dei termini integrali della matrice di cedibilità secondo
REM la formula 4.21. Termine (1,1)
REM -----
CALL Plv(M1(), M1(), N1(), N1(), T1(), T1(), C(1, 1))
REM -----

```



```

REM Costruzione degli altri due termini della matrice di cedibilita' secondo
REM la formula 4.21
REM -----
CALL Imposte(A(), B(), R1(), R1(), P1(), P1(), C(1, 1))
REM -----
REM Costruzione del termine (2,2) della matrice di cedibilita'
REM -----
CALL Plv(M2(), M2(), N2(), N2(), T2(), T2(), C(2, 2))
CALL Imposte(A(), B(), R2(), R2(), P2(), P2(), C(2, 2))
REM -----
REM Costruzione del termine (3,3) della matrice di cedibilita'
REM -----
CALL Plv(M3(), M3(), N3(), N3(), T3(), T3(), C(3, 3))
CALL Imposte(A(), B(), R3(), R3(), P3(), P3(), C(3, 3))
REM -----
REM Costruzione del termine (1,2) della matrice di cedibilita'
REM -----
CALL Plv(M1(), M2(), N1(), N2(), T1(), T2(), C(1, 2))
CALL Imposte(A(), B(), R1(), R2(), P1(), P2(), C(1, 2))
REM -----
REM Costruzione del termine (1,3) della matrice di cedibilita'
REM -----
CALL Plv(M1(), M3(), N1(), N3(), T1(), T3(), C(1, 3))
CALL Imposte(A(), B(), R1(), R3(), P1(), P3(), C(1, 3))
REM -----
REM Costruzione del termine (2,3) della matrice di cedibilita'
REM -----
CALL Plv(M2(), M3(), N2(), N3(), T2(), T3(), C(2, 3))
CALL Imposte(A(), B(), R2(), R3(), P2(), P3(), C(2, 3))
REM -----
REM Imposizione della simmetria
REM -----
C(2, 1) = C(1, 2): C(3, 1) = C(1, 3): C(3, 2) = C(2, 3)
END SUB
SUB dot (A(), B(), C, N)
REM
REM *****
REM *
REM *
REM *
REM *
REM *****
REM

```

```

REM Prodotto scalare tra due vettori di ordine N
REM -----
C = 0
FOR I = 1 TO N
  C = C + A(I) * B(I)
NEXT I
END SUB
SUB Geometria
REM
REM *****
REM *
REM *           G e o m e t r i a
REM *
REM *****
REM
REM -----
REM Riempimento degli array delle aree, delle inerzie e dei fattori di taglio
REM -----
FOR I = 1 TO T + 1
  Z = LUCE / T * (I - 1)
  AREA(I) = FNA(Z) * FNB(Z) - (FNA(Z) - FNC(Z)) * (FNB(Z) - 2 * FNS(Z))
  INERZ(I) = FNA(Z) * FNB(Z) ^ 3 / 12 - (FNA(Z) - FNC(Z)) * (FNB(Z) - 2 *
    FNS(Z)) ^ 3 / 12
  CHI(I) = AREA(I) / (FNC(Z) * (FNB(Z) - 2 * FNS(Z)))
NEXT I
REM -----
REM Riempimento dell'array delle quote, se descritte da funzioni
REM -----
IF ASSE$ = "FUNC" THEN
  FOR I = 1 TO T + 1
    Z = LUCE / T * (I - 1)
    Q(I) = FNQ(Z)
  NEXT I
END IF
REM -----
REM Riempimento degli array dei carichi, se descritti da funzioni
REM -----
IF CARICHI$ = "FUNC" THEN
  FOR I = 2 TO T
    Z1 = LUCE / T * (I - 1) - LUCE / T / 2
    Z2 = LUCE / T * I - LUCE / T / 2
    FVER(I) = (FNF(Z1) + FNF(Z2)) / 2 * LUCE / T
  
```

```

FORI(I) = (FNP(Z1) + FNP(Z2)) / 2 * LUCE / T
NEXT I
FVER(1) = 0: FVER(T + 1) = 0: FORI(1) = 0: FORI(T + 1) = 0
END IF
REM -----
REM Riempimento degli array delle inclinazioni dei tratti
REM -----
INCL(1) = ATN(Q(2) / LUCE * T)
FOR I = 2 TO T
  INCL(I) = ATN((Q(I + 1) - Q(I - 1)) / LUCE / 2 * T)
NEXT I
INCL(T + 1) = ATN((Q(T + 1) - Q(T)) / LUCE * T)
END SUB
SUB Imposte (A(), B(), RA(), RB(), PA(), PB(), C)
REM
REM *****
REM *
REM *
REM *
REM *
REM *****
DIM U(3) AS DOUBLE
REM -----
REM Opera il prodotto A per RA
REM -----
CALL product(A(), RA(), U(), 3)
REM -----
REM Opera il prodotto interno RB · (A RA)
REM -----
CALL dot(RB(), U(), CA, 3)
REM -----
REM Opera il prodotto B per PA
REM -----
CALL product(B(), PA(), U(), 3)
REM -----
REM Opera il prodotto interno PB · (B PA)
REM -----
CALL dot(PB(), U(), BA, 3)
C = C + CA + BA
END SUB
SUB Imposte1 (SA(), SB(), RA(), RB(), C)
REM
REM *****

```

```

REM *
REM *                               Imposte1
REM *
REM *****
DIM U(3) AS DOUBLE
REM -----
REM Opera il prodotto interno RA . SA
REM -----
CALL dot(RA(), SA(), CA, 3)
REM -----
REM Opera il prodotto interno RB . SB
REM -----
CALL dot(RB(), SB(), BA, 3)
C = C - CA - BA
END SUB
SUB IngressoDati
REM
REM *****
REM *
REM *                               I n g r e s s o D a t i
REM *
REM *****
REM
REM *****
REM *
REM *                               Questa subroutine si occupa dell'ingresso dei dati non
REM *                               organizzati in array
REM *
REM *****
REM *****
REM *
REM * LEGGE IL TITOLO DELLA STRUTTURA
REM * IL FLAG DI STAMPA PER LE VARIABILI DI INGRESSO
REM * IL FLAG DI STAMPA PER LE VARIABILI DI USCITA
REM * - CARTA → I DATI VENGONO STAMPATI
REM * - VIDEO → I DATI VENGONO INVIATI A VIDEO
REM * - FILES → I DATI VENGONO MEMORIZZATI SU
REM *          FILES DA SPECIFICARE
REM * - VIDEOFILES → I DATI VENGONO INVIATI A
REM *          VIDEO E MEMORIZZATI SU FILES
REM * - CARTAFILES → I DATI VENGONO STAMPATI
REM *          CARTA E MEMORIZZATI SU FILES

```



```

REM *      SPTD   Spessore complessivo dei timpani dell'arco a destra      *
REM *      A(3,3) Matrice (3,3) dei cedimenti elastici della                *
REM *              imposta di sinistra                                     *
REM *      B(3,3) Matrice (3,3) dei cedimenti elastici della                *
REM *              imposta di destra                                       *
REM *      SA(3)  Matrice (3) dei cedimenti anelastici della                *
REM *              imposta di sinistra                                     *
REM *      SB(3)  Matrice (3) dei cedimenti anelastici della                *
REM *              imposta di destra                                       *
REM *      ASSE$  Se l'asse dell'arco e' descritto da una                  *
REM *              funzione, ASSE$ = FUNZ, se invece e' dato                *
REM *              per punti, allora ASSE$ = DATI                          *
REM *      CARICHI$ Se i carichi sull'arco sono descritti da                *
REM *              una funzione, CARICHI$ = FUNZ, se invece                  *
REM *              sono definiti per punti, allora CARICHI$ = DATI          *
REM *      Se l'asse e' definito da funzione:                               *
REM *      FREC   Freccia dell'arco                                         *
REM *      RIAL   Rialzo a destra                                           *
REM *      Se i carichi sono definiti da funzione:                           *
REM *      FVS   Carico verticale all'estremo di sinistra                   *
REM *      FVC   Carico verticale al centro                                 *
REM *      FVD   Carico verticale all'estremo di destra                     *
REM *      FOS   Carico orizzontale all'estremo di sinistra                 *
REM *      FOC   Carico orizzontale al centro                               *
REM *      FOD   Carico orizzontale all'estremo di destra                   *
REM *
REM *****
REM
INPUT #7, LUCE, T
INPUT #7, E, NU
INPUT #7, ALPHA, DELTATN, DELTATU
INPUT #7, LARS, LARC, LARD, ALTS, ALTC, ALTD
INPUT #7, SPES, SPEC, SPED, SPTS, SPTC, SPTD
FOR I = 1 TO 3
  FOR J = 1 TO 3
    INPUT #7, A(I, J)
  NEXT J
NEXT I
FOR I = 1 TO 3
  FOR J = 1 TO 3
    INPUT #7, B(I, J)
  NEXT J

```

```

NEXT I
FOR I = 1 TO 3
  INPUT #7, SA(I)
NEXT I
FOR I = 1 TO 3
  INPUT #7, SB(I)
NEXT I
INPUT #7, ASSE$
INPUT #7, CARICHI$
IF ASSE$ = "FUNC" THEN
  INPUT #7, FREC, RIAL
END IF
IF CARICHI$ = "FUNC" THEN
  INPUT #7, FVS, FVC, FVD, FOS, FOC, FOD
END IF
END SUB
SUB IngressoDatiArray
REM
REM *****
REM *
REM *           I n g r e s s o D a t i A r r a y
REM *
REM *****
REM
REM *****
REM *
REM *           Questa subroutine si occupa dell'ingresso dei dati
REM *           organizzati in array
REM *
REM *****
REM
REM -----
REM Legge le quote dell'arco
REM -----
IF ASSE$ = "DATI" THEN
  FOR I = 1 TO T + 1
    INPUT #7, Q(I)
  NEXT I
  RIAL = Q(T + 1)
  FREC = Q(T / 2 + 1) - RIAL / 2
END IF
REM

```

```

REM *****
REM *
REM *           Legge i carichi verticali ed orizzontali
REM *
REM *****
IF CARICHI$ = "DATI" THEN
  FOR I = 1 TO T + 1
    INPUT #7, FVER(I)
  NEXT I
  FOR I = 1 TO T + 1
    INPUT #7, FORI(I)
  NEXT I
END IF
END SUB
SUB Iperstatiche
REM
REM *****
REM *
REM *           I p e r s t a t i c h e
REM *
REM *****
REM
REM
REM *****
REM *
REM * QUESTA SUBROUTINE CALCOLA LE INCOGNITE IPERSTATICHE, *
REM *           RISOLVENDO IL SISTEMA DI FLESSIBILITA'
REM *
REM *****
REM
REM _____
REM Chiama la subroutine SLSIM per la soluzione alla Gauss del sistema
REM _____
CALL slsim(C(), BB(), X(), 3)
END SUB
SUB Plv (MA(), MB(), NA(), NB(), TA(), TB(), C)
REM
REM *****
REM *
REM *           P l v
REM *
REM *****

```


170 *Le strutture ad arco*

```

C = 0
FOR I = 1 TO T
  MA = (MA(I) + MA(I + 1)) / 2
  MB = (MB(I) + MB(I + 1)) / 2
  NA = (NA(I) + NA(I + 1)) / 2
  NB = (NB(I) + NB(I + 1)) / 2
  TA = (TA(I) + TA(I + 1)) / 2
  TB = (TB(I) + TB(I + 1)) / 2
  AREA = (AREA(I) + AREA(I + 1)) / 2
  INERZ = (INERZ(I) + INERZ(I + 1)) / 2
  CHI = (CHI(I) + CHI(I + 1)) / 2
  LUNGH = SQR((Q(I + 1) - Q(I)) ^ 2 + (LUCE / T) ^ 2)
  C = C + LUNGH * MA * MB / E / INERZ
  C = C + LUNGH * NA * NB / E / AREA
  C = C + LUNGH * CHI * TA * TB / G / AREA
NEXT I
END SUB
SUB Plv1 (MA(), MB(), NA(), NB(), TA(), TB(), C)
REM
REM *****
REM *
REM *
REM *
REM *
REM *****
C = 0
FOR I = 1 TO T
  MA = (MA(I) + MA(I + 1)) / 2
  MB = (MB(I) + MB(I + 1)) / 2
  NA = (NA(I) + NA(I + 1)) / 2 - FVER(I + 1) * SIN(INCL(I + 1)) / 2
  NA = NA + FORI(I + 1) * COS(INCL(I + 1)) / 2
  NB = (NB(I) + NB(I + 1)) / 2
  TA = (TA(I) + TA(I + 1)) / 2 + FVER(I + 1) * COS(INCL(I + 1)) / 2
  TA = TA + FORI(I + 1) * SIN(INCL(I + 1)) / 2
  TB = (TB(I) + TB(I + 1)) / 2
  AREA = (AREA(I) + AREA(I + 1)) / 2
  INERZ = (INERZ(I) + INERZ(I + 1)) / 2
  CHI = (CHI(I) + CHI(I + 1)) / 2
  LUNGH = SQR((Q(I + 1) - Q(I)) ^ 2 + (LUCE / T) ^ 2)
  C = C + LUNGH * MA * MB / E / INERZ
  C = C + LUNGH * NA * NB / E / AREA
  C = C + LUNGH * CHI * TA * TB / G / AREA
REM

```

```

REM Effetto delle variazioni termiche
REM -----
Z = LUCE / T * (I - 1)
Z1 = LUCE / T * I
HH = (FNB(Z) + FNB(Z1)) / 2
C = C + MB * LUNGH * ALPHA * DELTATN / HH
C = C + NB * ALPHA * DELTATU * LUNGH
NEXT I
END SUB
SUB product (A(), B(), C(), N)
REM
REM *****
REM *
REM *                product
REM *
REM *****
FOR I = 1 TO N
  C(I) = 0
  FOR K = 1 TO N
    C(I) = C(I) + A(I, K) * B(K)
  NEXT K
NEXT I
END SUB
SUB slsim (A(), B(), X(), N)
REM
REM *****
REM *
REM *                s l s i m
REM *
REM *****
REM
REM
REM *****
REM *
REM *  QUESTA SUBROUTINE RISOLVE UN SISTEMA DI EQUAZIONI
REM *    LINEARI DI ORDINE N, A MATRICE SIMMETRICA,
REM *    TRAMITE IL METODO DI GAUSS. NON E' SUPPOSTA BANDA.
REM *
REM *****
REM
REM
FOR K = 1 TO N - 1
  C = A(K, K)

```

```

FOR J = K + 1 TO N
  A(J, K) = A(K, J)
NEXT J
IF ABS(C) < 1E-10 THEN
  CLS
  PRINT "MATRICE MAL CONDIZIONATA O SINGOLARE"
  SLEEP
END IF
FOR J = K + 1 TO N
  A(K, J) = A(K, J) / C
NEXT J
B(K) = B(K) / C
FOR I = K + 1 TO N
  C = A(I, K)
  FOR J = I TO N
    A(I, J) = A(I, J) - C * A(K, J)
  NEXT J
  B(I) = B(I) - C * B(K)
NEXT I
NEXT K
IF ABS(A(N, N)) < 1E-10 THEN
  CLS
  PRINT "MATRICE MAL CONDIZIONATA O SINGOLARE"
  SLEEP
END IF
B(N) = B(N) / A(N, N)
FOR L = 1 TO N - 1
  K = N - L
  FOR J = K + 1 TO N
    B(K) = B(K) - A(K, J) * B(J)
  NEXT J
NEXT L
FOR I = 1 TO 3: X(I) = -B(I): NEXT I
END SUB
SUB Spostamenti
REM
REM *****
REM *
REM *           S p o s t a m e n t i
REM *
REM *****
REM

```

```

REM *****
REM *
REM *   QUESTA SUBROUTINE CALCOLA SPOSTAMENTI E ROTAZIONI *
REM *
REM *****
REM
DIM U(3), S3(3)
FOR I = 1 TO T
  M = (M(I) + M(I + 1)) / 2
  N = (N(I) + N(I + 1) - FVER(I + 1) * SIN(INCL(I + 1))) / 2
  N = N + (FORI(I + 1) * COS(INCL(I + 1))) / 2
  TG = (TG(I) + TG(I + 1) + FVER(I + 1) * COS(INCL(I + 1))) / 2
  TG = TG + (FORI(I + 1) * SIN(INCL(I + 1))) / 2
  AREA = (AREA(I) + AREA(I + 1)) / 2
  INERZ = (INERZ(I) + INERZ(I + 1)) / 2
  PEND = (INCL(I) + INCL(I + 1)) / 2
  HH = (FNB(I + 1) + FNB(I)) / 2
  QQ = Q(I + 1) - Q(I)
  CHI = (CHI(I) + CHI(I + 1)) / 2
  LUNGH = SQR((Q(I + 1) - Q(I)) ^ 2 + (LUCE / T) ^ 2)
  REM Cfr. (4.54)
  V(I + 1) = V(I) - (M / E / INERZ + ALPHA * DELTATN / HH) * LUNGH *
    LUCE / T / 2
  V(I + 1) = V(I + 1) - (N / E / AREA + ALPHA * DELTATU) * LUNGH *
    SIN(PEND)
  V(I + 1) = V(I + 1) + CHI * TG / G / AREA * LUNGH * COS(PEND)
  V(I + 1) = V(I + 1) - PHI(I) * LUCE / T
  REM Cfr. (4.55)
  W(I + 1) = W(I) - (M / E / INERZ + ALPHA * DELTATN / HH) * LUNGH *
    QQ / 2
  W(I + 1) = W(I + 1) + (N / E / AREA + ALPHA * DELTATU) * LUNGH *
    COS(PEND)
  W(I + 1) = W(I + 1) + CHI * TG / G / AREA * LUNGH * SIN(PEND)
  W(I + 1) = W(I + 1) - PHI(I) * QQ
  REM Cfr. (4.56)
  PHI(I + 1) = PHI(I) + M / E / INERZ * LUNGH + ALPHA * DELTATN / HH
    * LUNGH
NEXT I
REM Cfr. (4.52)
FOR I = 1 TO 3
  R(I) = R0(I) + X(1) * R1(I) + X(2) * R2(I) + X(3) * R3(I)
  P(I) = P0(I) + X(1) * P1(I) + X(2) * P2(I) + X(3) * P3(I)

```

```

NEXT I
CALL product(A(), R(), U(), 3)
FOR I = 1 TO 3: S3(I) = SA(I) - U(I): NEXT I
REM Cfr. (4.53)
FOR I = 1 TO T + 1
  Z = LUCE / T * (I - 1)
  V(I) = V(I) + S3(1) - S3(3) * Z
  W(I) = W(I) + S3(2) - S3(3) * Q(I)
  PHI(I) = PHI(I) + S3(3)
NEXT I
END SUB
SUB TerminiNoti
REM
REM *****
REM *
REM *           T e r m i n i N o t i
REM *
REM *****
REM
REM *****
REM *
REM *           Q U E S T A   S U B R O U T I N E   C A L C O L A   I   T E R M I N I   N O T I
REM *           D E L L E   E Q U A Z I O N I   D I   C O N G R U E N Z A
REM *
REM *****
REM
REM _____
REM Reazioni a destra ed a sinistra, secondo le (4.37) e (4.43)
REM _____
VA = 0: VB = 0: HA = 0: HB = 0
FOR I = 1 TO T / 2 + 1
  Z = LUCE / T * (I - 1)
  VA = VA - FVER(I) * (1 - (2 * FREC - RIAL) / 2 / FREC / LUCE * Z)
  VA = VA + FORI(I) * Q(I) * (2 * FREC - RIAL) / 2 / FREC / LUCE
  HA = HA + FVER(I) * Z / 2 / FREC - FORI(I) * (2 * FREC - Q(I)) / 2 / FREC
  VB = VB - FVER(I) * (2 * FREC - RIAL) / 2 / FREC / LUCE * Z
  VB = VB - FORI(I) * Q(I) * (2 * FREC - RIAL) / 2 / FREC / LUCE
  HB = HB - FVER(I) * Z / 2 / FREC - FORI(I) * Q(I) / 2 / FREC
NEXT I
REM _____
REM Reazioni a destra ed a sinistra, secondo le (4.39) e (4.45)
REM _____

```

```

FOR I = T / 2 + 2 TO T + 1
  Z = LUCE / T * (I - 1)
  VA = VA - FVER(I) * (2 * FREC + RIAL) / 2 / FREC / LUCE * (LUCE - Z)
  VA = VA + FORI(I) * (Q(I) - RIAL) * (2 * FREC + RIAL) / 2 / FREC / LUCE
  HA = HA + FVER(I) * (LUCE - Z) / 2 / FREC
  HA = HA - FORI(I) * (Q(I) - RIAL) / 2 / FREC
  VB = VB - FVER(I) * ((2 * FREC + RIAL) * Z - RIAL * LUCE) / 2 / FREC /
    LUCE
  VB = VB - FORI(I) * (Q(I) - RIAL) * (2 * FREC + RIAL) / 2 / FREC / LUCE
  HB = HB - FVER(I) * (LUCE - Z) / 2 / FREC
  HB = HB - FORI(I) * (2 * FREC - Q(I) + RIAL) / 2 / FREC
NEXT I
REM -----
REM Momenti, tagli e sforzi normali, secondo le (4.40) e (4.46)
REM -----
FOR I = 1 TO T + 1
  Z = LUCE / T * (I - 1)
  M0(I) = M0(I) - VA * Z - HA * Q(I)
  N0(I) = N0(I) + VA * SIN(INCL(I)) - HA * COS(INCL(I))
  T0(I) = T0(I) - VA * COS(INCL(I)) - HA * SIN(INCL(I))
NEXT I
FOR I = 1 TO T + 1
  FOR J = 1 TO I
    Z = LUCE / T * (I - 1)
    Z1 = LUCE / T * (J - 1)
    M0(I) = M0(I) - FVER(J) * (Z - Z1) - FORI(J) * (Q(I) - Q(J))
    N0(I) = N0(I) + FVER(J) * SIN(INCL(I)) - FORI(J) * COS(INCL(I))
    T0(I) = T0(I) - FVER(J) * COS(INCL(I)) - FORI(J) * SIN(INCL(I))
  NEXT J
NEXT I
REM -----
REM Matrici delle reazioni
REM -----
R0(1) = VA: R0(2) = HA: R0(3) = 0
P0(1) = VB: P0(2) = HB: P0(3) = 0
REM -----
REM Costruzione del primo termine noto, secondo la (4.34)
REM -----
REM Termini integrali della (4.34)
CALL Plv1(M0(), M1(), N0(), N1(), T0(), T1(), BB(1))
REM Primo e secondo termine della (4.34)
CALL Imposte(A(), B(), R0(), R1(), P0(), P1(), BB(1))

```

```

REM Terzo e quarto termine della (4.34)
CALL Impostel(SA(), SB(), R1(), P1(), BB(1))
REM -----
REM Costruzione del secondo termine noto
REM -----
CALL Plv1(M0(), M2(), N0(), N2(), T0(), T2(), BB(2))
CALL Imposte(A(), B(), R0(), R2(), P0(), P2(), BB(2))
CALL Impostel(SA(), SB(), R2(), P2(), BB(2))
REM -----
REM Costruzione del terzo termine noto
REM -----
CALL Plv1(M0(), M3(), N0(), N3(), T0(), T3(), BB(3))
CALL Imposte(A(), B(), R0(), R3(), P0(), P3(), BB(3))
CALL Impostel(SA(), SB(), R3(), P3(), BB(3))
END SUB
SUB UscitaDati
REM
REM *****
REM *
REM *          U s c i t a D a t i          *
REM *
REM *****
REM
REM *****
REM *
REM *          Questa subroutine si occupa dell'uscita (su video, su stampa
REM *          e su file) dei dati di ingresso
REM *
REM *****
FOR ITER = 1 TO ITER1
  PRINT #ITER, " ===== "
  PRINT #ITER, "PROGRAMMA ADL - ANALISI ELASTICA"
  PRINT #ITER, TITLE$
  PRINT #ITER, " ===== "
  PRINT #ITER,
  PRINT #ITER, "FILE DEI DATI ="; FILEINPUT$
  PRINT #ITER, "LUCE DELL'ARCO = "; LUCE
  PRINT #ITER, "FRECCIA DELL'ARCO = "; FREC
  PRINT #ITER, "RIALZO A DESTRA ="; RIAL
  PRINT #ITER, "MODULO DI YOUNG ="; E
  PRINT #ITER, "MODULO DI POISSON ="; NU
  PRINT #ITER, "COEFFICIENTE DI DILATAZIONE TERMICA ="; ALPHA

```

```

PRINT #ITER, "VARIAZIONE TERMICA ALLA NAVIER ="; DELTATN
PRINT #ITER, "VARIAZIONE TERMICA UNIFORME ="; DELTATU
PRINT #ITER, "NUMERO DI TRATTI = "; T
PRINT #ITER, "LARGHEZZA DELLA SEZIONE RETTA DELL'ARCO A SIN-
  ISTRA = "; LARS
PRINT #ITER, "LARGHEZZA DELLA SEZIONE RETTA DELL'ARCO IN CHI-
  AVE = "; LARC
PRINT #ITER, "LARGHEZZA DELLA SEZIONE RETTA DELL'ARCO A DE-
  STRA = "; LARD
PRINT #ITER, "ALTEZZA DELLA SEZIONE RETTA DELL'ARCO A SINIS-
  TRA = "; ALTS
PRINT #ITER, "ALTEZZA DELLA SEZIONE RETTA DELL'ARCO IN CHIAVE
  = "; ALTC
PRINT #ITER, "ALTEZZA DELLA SEZIONE RETTA DELL'ARCO A DESTRA
  = "; ALTD
PRINT #ITER, "SPESSORE DELLA SOLETTA A SINISTRA = "; SPES
PRINT #ITER, "SPESSORE DELLA SOLETTA AL CENTRO = "; SPEC
PRINT #ITER, "SPESSORE DELLA SOLETTA A DESTRA = "; SPED
PRINT #ITER, "SPESSORE COMPLESSIVO DELLE NERVATURE A SINIS-
  TRA = "; SPTS
PRINT #ITER, "SPESSORE COMPLESSIVO DELLE NERVATURE AL CEN-
  TRO = "; SPTC
PRINT #ITER, "SPESSORE COMPLESSIVO DELLE NERVATURE A DESTRA
  = "; SPTD
PRINT #ITER,
PRINT #ITER, "MATRICE DELLE CEDIBILITA' ELASTICHE A SINISTRA"
PRINT #ITER,
FOR I = 1 TO 3
  PRINT #ITER, TAB(2); A(I, 1); TAB(20); A(I, 2); TAB(45); A(I, 3)
NEXT I
PRINT #ITER,
PRINT #ITER, "MATRICE DELLE CEDIBILITA' ELASTICHE A DESTRA"
PRINT #ITER,
FOR I = 1 TO 3
  PRINT #ITER, TAB(2); B(I, 1); TAB(20); B(I, 2); TAB(45); B(I, 3)
NEXT I
PRINT #ITER,
PRINT #ITER, "VETTORE DEI CEDIMENTI ANELASTICI A SINISTRA"
PRINT #ITER,
PRINT #ITER, TAB(2); SA(1); TAB(20); SA(2); TAB(45); SA(3)
PRINT #ITER,
PRINT #ITER, "VETTORE DEI CEDIMENTI ANELASTICI A DESTRA"

```



```

PRINT #ITER,
PRINT #ITER, TAB(2); SB(1); TAB(20); SB(2); TAB(45); SB(3)
PRINT #ITER,
IF CARICHI$ = "FUNC" THEN
  PRINT #ITER, "CARICO VERTICALE ALL'ESTREMO DI SINISTRA = ";
  FVS
  PRINT #ITER, "CARICO VERTICALE AL CENTRO = "; FVC
  PRINT #ITER, "CARICO VERTICALE ALL'ESTREMO DI DESTRA = ";
  FVD
  PRINT #ITER, "CARICO ORIZZONTALE ALL'ESTREMO DI SINISTRA =
"; FOS
  PRINT #ITER, "CARICO ORIZZONTALE AL CENTRO = "; FOC
  PRINT #ITER, "CARICO ORIZZONTALE ALL'ESTREMO DI DESTRA = ";
  FOD
  PRINT #ITER,
END IF
REM -----
REM STAMPA LE QUOTE DELL'ARCO PER CIASCUNA DIVIDENTE
REM -----
PRINT #ITER, " ===== "
PRINT #ITER, " = "
PRINT #ITER, " =          TABELLA DELLE QUOTE DELL'ARCO          = "
PRINT #ITER, " = "
PRINT #ITER, " ===== "
PRINT #ITER,
PRINT #ITER, " ----- "
PRINT #ITER, " DIVIDENTE          QUOTA          "
PRINT #ITER, " ----- "
PRINT #ITER,
FOR I = 1 TO T + 1
  PRINT #ITER, TAB(1); I; TAB(34); Q(I)
NEXT I
REM -----
REM STAMPA I CARICHI PER CIASCUNA DIVIDENTE
REM -----
PRINT #ITER, " ===== "
PRINT #ITER, " = "
PRINT #ITER, " = CARICHI VERTICALI ED ORIZZONTALI SULL'ARCO = "
PRINT #ITER, " = "
PRINT #ITER, " ===== "
PRINT #ITER,
PRINT #ITER, " ----- "

```

```

PRINT #ITER, " DIVIDENTE CAR.VERTICALI CAR.ORIZZONTALE "
PRINT #ITER, "                 "
PRINT #ITER,
FOR I = 1 TO T + 1
  PRINT #ITER, TAB(2); I; TAB(15); FVER(I); TAB(35); FORI(I)
NEXT I
NEXT ITER
END SUB
SUB UscitaRisultati
REM
REM *****
REM *
REM *          U s c i t a R i s u l t a t i
REM *
REM *****
REM
REM
REM *****
REM *
REM *          Questa subroutine si occupa dell'uscita (su video, su stampa
REM *
REM *          e su file) dei risultati
REM *
REM *****
REM
FOR ITER = ITER1 + 1 TO ITER2
  PRINT #ITER, "REAZIONE VERTICALE A SINISTRA ="; R(1)
  PRINT #ITER, "REAZIONE ORIZZONTALE A SINISTRA ="; R(2)
  PRINT #ITER, "REAZIONE VERTICALE A DESTRA ="; P(1)
  PRINT #ITER, "REAZIONE ORIZZONTALE A DESTRA ="; P(2)
  SUMF = 0: SUMO = 0
  FOR I = 1 TO T + 1
    SUMF = SUMF + FVER(I)
    SUMO = SUMO + FORI(I)
  NEXT I
  PRINT #ITER, "SOMMA DELLE FORZE VERTICALI ="; SUMF
  PRINT #ITER, "SOMMA DELLE FORZE ORIZZONTALI ="; SUMO
  PRINT #ITER, "CARICO VERTICALE MEDIO ="; SUMF / LUCE
  PRINT #ITER, "CARICO ORIZZONTALE MEDIO ="; SUMO / LUCE
  PRINT #ITER, " ===== "
  PRINT #ITER, " = "
  PRINT #ITER, " =          MOMENTI FLETTENTI          = "
  PRINT #ITER, " = "

```

```

PRINT #ITER, " ====="
PRINT #ITER,
PRINT #ITER, " ====="
PRINT #ITER, " DIVIDENTE          MOMENTO FLETTENTE "
PRINT #ITER, " ====="
PRINT #ITER,
FOR I = 1 TO T + 1
  PRINT #ITER, TAB(2); I; TAB(15); M(I)
NEXT I
PRINT #ITER, " ====="
PRINT #ITER, " = "
PRINT #ITER, " =          SFORZI NORMALI          = "
PRINT #ITER, " = "
PRINT #ITER, " ====="
PRINT #ITER,
PRINT #ITER, " ====="
PRINT #ITER, " DIVIDENTE          N A SINISTRA          N A DESTRA "
PRINT #ITER, " ====="
PRINT #ITER,
FOR I = 1 TO T + 1
  PRINT #ITER, TAB(2); I; TAB(10); NS(I); TAB(35); N(I)
NEXT I
PRINT #ITER, " ====="
PRINT #ITER, " = "
PRINT #ITER, " =          TAGLIO          = "
PRINT #ITER, " = "
PRINT #ITER, " ====="
PRINT #ITER,
PRINT #ITER, " ====="
PRINT #ITER, " DIVIDENTE          T A SINISTRA          T A DESTRA "
PRINT #ITER, " ====="
PRINT #ITER,
FOR I = 1 TO T + 1
  PRINT #ITER, TAB(2); I; TAB(10); TS(I); TAB(35); TG(I)
NEXT I
PRINT #ITER, " ====="
PRINT #ITER, " = "
PRINT #ITER, " =          SPOSTAMENTI          = "
PRINT #ITER, " = "
PRINT #ITER, " ====="
PRINT #ITER,
PRINT #ITER, " ====="

```

```

PRINT #ITER, " DIVIDENTE SPOS. VERTICALI SPOS. ORIZZONTALI "
PRINT #ITER, " ===== "
PRINT #ITER,
FOR I = 1 TO T + 1
  PRINT #ITER, TAB(2); I; TAB(10); V(I); TAB(35); W(I)
NEXT I
PRINT #ITER, " ===== "
PRINT #ITER, " = "
PRINT #ITER, " = "
PRINT #ITER, " = "
PRINT #ITER, " = "
PRINT #ITER, " ===== "
PRINT #ITER,
PRINT #ITER, " ===== "
PRINT #ITER, " TRATTO "
PRINT #ITER, " = "
PRINT #ITER, " ===== "
PRINT #ITER,
FOR I = 1 TO T + 1
  PRINT #ITER, TAB(2); I; TAB(15); PHI(I)
NEXT I
PRINT #ITER,
PRINT #ITER, "TEMPO DI ESECUZIONE = "; TIME2; " SEC."
PRINT #ITER, "RUN DEL "; DATE$; " ORE "; TIME$
NEXT ITER
END SUB

```